

AMENDMENTS TO THE CLAIMS

Please amend the claims as indicated in the following listing of all claims:

1. (Cancelled)
2. (Currently amended) A method ~~for operating a distributed arbiter for a plurality of resources~~, comprising:  
receiving at one resource of ~~the~~ a plurality of resources, requests for the one resource from a plurality of requesters;  
granting the one resource to one of the requesters according to respective requester priorities, the respective requester priorities being inversely related to a number of requests made respectively, by the requesters; and  
the requesters supplying respective requester priority indications to at least the one resource.
3. (Original) The method as recited in claim 2 wherein requesters respectively supply number of requests as requester priority indications.
4. (Previously presented) The method as recited in claim 2 further comprising granting at least one of the plurality of resources to a requester according to a round robin scheme, thereby avoiding starvation.
5. (Previously presented) The method as recited in claim 2 wherein the one resource provides a grant indication to the one of the plurality of requesters.
6. (Currently amended) The method as recited in claim 2 wherein the ~~multiple~~ plurality of resources evaluate received requests sequentially.
7. (Original) The method as recited in claim 6 wherein resources are considered in an order determined according to resource priority.

8. (Currently amended) The method as recited in claim 2 wherein the ~~multiple~~ plurality of resources evaluate received requests sequentially, and wherein a requester already granted a resource removes requests for other resources for arbitration for a next sequential resource, thereby preventing requesters from receiving multiple grants during an arbitration cycle.

9. (Previously presented) The method as recited in claim 2 wherein requester priorities are recalculated after a resource is granted.

10. (Currently amended) The method as recited in claim 2 further comprising ~~the distributed-arbiter~~ performing multiple iterations during one arbitration cycle to allocate resources to requesters.

11. (Original) The method as recited in claim 9 further comprising requesters recalculating their requests by eliminating requests for a granted resource.

12. (Currently amended) A method ~~for operating a distributed-arbiter for a plurality of resources~~, comprising:  
 receiving at one resource ~~of the a plurality of~~ resources, requests for the one resource from a plurality of requesters; and  
 granting the one resource to one of the requesters according to respective requester priorities, the respective requester priorities being inversely related to a number of requests made respectively, by the requesters,  
 wherein the one resource sends a resource priority indication to the requester being granted its request, indicative of a number of requests received by the resource,  
 and  
 wherein requesters receive multiple grants from multiple resources during an arbitration iteration.

13. (Original) The method as recited in claim 12 wherein a number of arbitration iterations over multiple arbitration cycles varies dynamically.

14. (Original) The method as recited in claim 12 wherein the one resource determines which requests to grant in parallel with other resources determining which requests to grant.

15. (Original) The method as recited in claim 12 wherein the one resource sends a grant indication to a requester being granted its request.

16. (Cancelled)

17. (Currently amended) A method ~~for operating a distributed arbiter for a plurality of resources~~, comprising:  
receiving at one resource ~~the~~ of a plurality of resources, requests for the one resource from a plurality of requesters; and  
granting the one resource to one of the requesters according to respective requester priorities, the respective requester priorities being inversely related to a number of requests made respectively, by the requesters, wherein requesters receive multiple grants from multiple resources during an arbitration iteration and  
wherein a requester selects from among a plurality of grants according to respective resource priorities associated with each granting resource, the respective resource priorities being inversely related to a number of requests made, respectively, for each granting resource.

18. (Original) The method as recited in claim 17 wherein resource priorities are recalculated each iteration.

19. (Original) The method as recited in claim 17 wherein requester priorities are recalculated each iteration.

20. (Original) The method as recited in claim 17 further comprising the requester utilizing a round robin scheme to select from among the plurality of grants.

21. (Original) The method as recited in claim 20 wherein the round robin scheme is utilized before the requester selects from among the plurality of grants according to the respective resource priorities, the requester selecting from among the plurality of grants according to the respective resource priorities if the round robin scheme does not result in an acceptance of a grant.

22. (Original) The method as recited in claim 17 further comprising the requester utilizing a random scheme in addition to using calculated priorities to select from among the plurality of grants.

23. (Original) The method as recited in claim 12 wherein the requester being granted its request sends an accept indication to the resource whose grant it is accepting.

24. (Original) The method as recited in claim 23 wherein a requester determines which of one or more grants to accept, in parallel with other requesters determining which received grants to accept.

25. (Previously presented) The method as recited in claim 2 wherein the resources do not transmit grant indications to requesters, each requester determining grant values according to received information from other requesters.

26. (Original) The method as recited in claim 25 wherein requesters and resources are synchronized in regards to round robin positions.

27. (Original) The method as recited in claim 2 wherein each of the requesters provides an indication of priority on a bus logically combining the indications of priority, the priority being related to a number of requests being made by each of the requesters.

28. (Original) The method as recited in claim 27 wherein the indications of priority are unary coded, one bit corresponding to one request.

29. (Original) The method as recited in claim 28 wherein the bus logically combines the indications of priority.

30. (Original) The method as recited in claim 28 wherein the indications of priority are a number of requests of respective requesters.

31. (Original) The method as recited in claim 28 further comprising a requester responding to the priority indication on the bus by not sending a request if the priority indication on the bus indicates a higher priority requester is requesting a resource.

32. (Previously presented) The method as recited in claim 2 wherein the requesters are nodes of a network coupled to input ports of a network switch and the resources are output ports of the network switch, multiple ones of the output ports being accessible to more than one of the input ports.

33. (Previously presented) The method as recited in claim 2 wherein the requesters are processors of a multi-processor system and the resources are memories coupled to the processors, each of the memories being accessible to more than one of the processors.

34. (Previously presented) The method as recited in claim 2 further comprising recalculating requester priorities after each time a resource is granted to a requester.

35. (Previously presented) An arbitration apparatus for arbitrating requests from a plurality of requesters for a plurality of resources, comprising:

means for receiving requests for resources from the requesters;

means for granting requests according to requester priorities, the requester priorities being inversely related to the number of requests respectively made by the requesters;

means for selecting from among a plurality of grants according to resource priorities, the resource priorities being inversely related to the number of requests made for the respective resource.

36. (Cancelled)

37. (Previously presented) The arbitration apparatus as recited in claim 35 further comprising means for preventing starvation for grants.

38. (Original) The arbitration apparatus as recited in claim 35 further comprising means for preventing starvation for requests.

39. (Currently amended) The arbitration apparatus as recited in claim 35 wherein the arbitration apparatus is one of a distributed arbiter and a centralized arbiter.

40. (Previously presented) A distributed arbiter comprising:  
a plurality of requesters;  
a plurality of resources coupled to the requesters through a transport mechanism;  
wherein each requester is coupled to provide to each resource requested by the respective requester, a request indication;  
wherein a requested resource is responsive to a plurality of requests to selectively grant one of the requests according to requester priorities, the requester priorities being inversely related to a number of requests being made by respective requesters; and  
wherein the requested resource is further coupled to provide an indication of resource priority, the resource priority being inversely related to a number of resource requests received by the requested resource.

41. (Original) The distributed arbiter as recited in claim 40 wherein the requested resource is further responsive to the plurality of requests to selectively grant one of the requests according to a round robin mechanism.

42. (Original) The method as recited in claim 41 wherein the round robin mechanism is utilized before the requested resource selects from among the plurality of requests according to the requester priorities, the requested resource granting one of the requests according to the requester priorities if the round robin mechanism does not result in a grant.

43. (Original) The distributed arbiter as recited in claim 40 wherein each requester requesting the requested resource respectively provides a requester priority indication to the requested resource.

44. (Original) The distributed arbiter as recited in claim 43 further comprising a bus coupled to the requesters and resources, and wherein each of the requesters provides an indication of priority on a bus, the bus logically combining the indications of priority, the priority being inversely related to a number of requests being made by each of the requesters.

45. (Original) The distributed arbiter as recited in claim 44 wherein the indications of priority are unary coded.

46. (Original) The distributed arbiter as recited in claim 44 wherein a bus implements a wired-NOR of the indications of priority.

47. (Original) The distributed arbiter as recited in claim 44 wherein each requester places a unary encoded number of requests on the bus as the indication of priority.

48. (Original) The method as recited in claim 44 wherein a requester with a lower priority than indicated on the bus is responsive to the priority indicated on the bus to not send its request to a resource.

49. (Original) The distributed arbiter as recited in claim 40 wherein the requested resource provides a grant indication to a requester whose request is being granted by the resource.

50. (Cancelled)

51. (Original) The distributed arbiter as recited in claim 40 wherein a requester is responsive to receiving a plurality of grants from a plurality of resources to accept one of the grants according to priorities of the grants, the priorities being inversely related to a number of requests received by resources.

52. (Original) The distributed arbiter as recited in claim 51 wherein the requester accepts a grant indicating a particular resource is being allocated to the requester, according to a starvation avoidance mechanism in addition to the priorities.

53. (Original) The distributed arbiter as recited in claim 40 wherein a requester is responsive to receiving a plurality of grants from a plurality of resources to accept one of the grants according to a random priority determined for the grants.

54. (Original) The distributed arbiter as recited in claim 40 wherein the requesters are nodes of a network coupled to input ports of a network switch and the resources are output ports of the network switch, multiple ones of the output ports being accessible to more than one of the input ports.

55. (Original) The distributed arbiter as recited in claim 40 wherein the requesters are processors of a multi-processor system and the resources are memories coupled to the processors, each of the memories being accessible to more than one of the processors.

56. (Currently amended) A computer program product encoded in at least one computer readable medium ~~to implement an arbitration mechanism to allocate a resource to requesters~~, the computer program product comprising:

a first instruction sequence executable to receive requests for one resource of ~~the~~ a plurality of resources from multiple requesters; and

a second instruction sequence executable to allocate the one resource to one of the requesters according to requester priorities, requester priorities being inversely related to a number of requests made by each requester for the resources;

a third instruction sequence executable to supply from the requesters, respective requester priority indications to at least the one resource; and

wherein the at least one computer readable medium is selected from the set of a disk, tape or other magnetic, optical, or electronic storage medium and a network, wireline, wireless or other communications medium.



57. (Original) The computer program product as recited in claim 56 wherein the computer program product further includes a starvation avoidance instruction sequence executable to implement a starvation avoidance mechanism.

58. (Original) A computer program product encoded in at least one computer readable medium to implement an arbitration mechanism to determine which of a plurality of grants received by a requester from a plurality of resources, to accept, the computer program product comprising:

a first instruction sequence executable to receive an indication of the grants for respective resources; and

a second instruction sequence executable to accept one of the grants according to grant priorities, grant priorities being inversely related to a number of requests received by a resource.

59. (Original) The computer program product as recited in claim 58 wherein the computer program product further includes a starvation avoidance instruction sequence executable to implement a starvation avoidance mechanism to accept one of the grants.

60. (Currently amended) The arbitration apparatus as recited in claim 35 further comprising:

means for logically combining indications of priority, the priority being related to a number of ~~request~~ requests being made by each of the requesters.

61. (New) The method as recited in claim 2 wherein the number of requests made respectively by the requesters is made to different ones of the plurality of resources.